UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/005,248	12/03/2001	Brian C. Barnes	2000.056500	7937

23720 7590 08/24/2006

WILLIAMS, MORGAN & AMERSON
10333 RICHMOND, SUITE 1100
HOUSTON, TX 77042

EXAMINER	
LEMMA, SAMSON B	

ART UNIT	PAPER NUMBER
2132	

DATE MAILED: 08/24/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/005,248	BARNES ET AL.	
	Examiner	Art Unit	
	Samson B. Lemma	2132	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *13 June 2006*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-24* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-24* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. This office action is in reply to an amendment filed on June 13, 2006 and **Claims 1-24** are pending/examined.

Response to Arguments

2. Applicant's argument filed on June 13, 2006 have been fully considered and they are not persuasive.

Applicant has not amended any of the independent claims. Thus Examiner found that the rejection made in the previous office action is also applicable to the argument presented by the applicant. Furthermore, Applicant's argument filed on June 13, 2006 is found to be similar to the argument filed on July 14, 2005, thus response presented by the examiner in the previous office action is also relevant and applicable towards the present applicant argument.

Referring to the independent claims 1, 9 and 17, it has been argued that, Draves, the reference on the record, does not teach restricting execution of security sensitive instructions by associating a first security identification (ID) with instructions and obtaining a second security ID associated with a software code (different than instruction(s)). Instead, Draves pairs two sets of handles with keys for a single or same item, i.e., a computer system resource shared between two different authorized processes (a server and client process). By using the two sets of handles with keys for the shared resource, Draves ensures that two different authorized processes can access that shared resource. In this manner, Draves does not use two security IDS associated two different items (the requested instruction(s) and the software code) for restricting the execution of security sensitive instructions. Based on the above indicated legal

standard, it is respectfully submitted that Draves fails to anticipate independent claims 1, 9. and 17.

Examiner disagrees with this argument, examiner would point out that Dravis on column 2, lines 27-31, disclose, the following. "The system provides for ensuring that a computer program is authorized to access a computer system resource. The system generates a system-wide resource table that has a resource entry **for each allocated resource. Each resource entry contains a preferably non-forgeable key that uniquely identifies the resource**. This indicates the fact that not only pairs keys are associated with both a single or same item, i.e., a computer system resource as but also associated **several resources and each resources/items** are also uniquely identified by the non-forgeable keys.

Examiner would also asserts that Dravis on column 3, lines 42-48, discloses the following, "In a preferred embodiment, the kernel maintains a system-wide resource table that is a hash table and that contains a resource entry corresponding **to each resource allocated by the kernel. The allocated resources are identified by a kernel-generated resource identifier.** The system of the present invention uses resource identifiers that contain both a handle and a key (a handle.backslash.key pair)."

This indicates the fact there is also several resources/items which are identified by the resource identifier or key pair.

Dravis further discloses the following, "When a process wishes to access the allocated resource, it passes the handle.backslash.key pair to the kernel. The kernel examines the resource entry indexed by the passed handle to determine whether the passed key is equal to the key in the indexed resource entry. The keys may not be equal for several reasons, including resource table compaction **and attempted forgery**." [Column 3, lines 63-column 4, line 2]. This implies that the requesting process could be any process including an unauthorized process which is attempting forgery however

forgery process is not able to access other resource that it is not authorized since it does not have the right key pair and the kernel denies this process from accessing the resources by matching the key with the resource it is requesting.

Examiner would point out Dravis in support of this discloses the following: "When no such resource entry is found, the kernel denies the process access to the resource. On the other hand, when a resource entry that contains a matching key is found, the kernel allows the process to access the resource." [Column 4, lines 7-10]

Finally examiner asserts that Dravis on column 3, lines 39-41, discloses the main feature of the invention indicating that it is directed to a method and system providing secure access to resources. The system provides for ensuring **that a computer program is authorized to access a computer system resource. And this implies that the invention is used to control access to any resources in the computer system by any computer programs.**

Most of applicant's argument presented on June 13, 2006, similar to that of the previous argument made on July 14, 2005, Examiner finds it useful to show how each limitation is disclosed by the reference on the record.

Therefore, in order to clarify how each and every limitation of the claim is disclosed by the reference on the record the examiner would show the independent claims 9, 1 and 17 as follows.

As per independent claims 9, 1 and 17 Draves discloses an apparatus, comprising:

- **A processor for running code thereon**,[column 3, lines 39-42 and column 1, lines 11-22 and figure 2, ref. Num "250"] (As indicated on column 3, lines 39-42, the invention is directed towards a method and system in a kernel of an operating system for providing secure access to computer system resources. The OS kernel is inherently operates in the processor. And as it is indicated on column 1, lines 39-42, the portion of the operating system that is

responsible for the allocation and deallocation of resources is known as the kernel. The kernel interacts with the shell and other programs as well as with the hardware devices on the system, including the processor (also called the central processing unit or CPU), memory and disk drives.)

- **For associating a first security identification (ID) with each of a plurality of instructions or a set of instructions that are to be executed by the processor;** [column 3, lines 43-50 and column 3, lines 60-62] (As it is disclosed on column 3, lines 60-62, each process which is defined as concurrently executing computer programs on column 1, lines 14-15, meets the limitation each of a plurality of instructions or a set of instructions are inherently executed by the processor are associated with the resource identifier comprising the handle/key pair that is passed to the process/programs/set of instructions when requesting allocation of resources. Furthermore Draves on column 3, lines 43-50 discloses the following. In a preferred embodiment, the kernel maintains a system-wide resource table that is a hash table and that contains a resource entry corresponding **to each resource allocated by the kernel. The allocated resources are identified by a kernel-generated resource identifier.** The system of the present invention uses resource identifiers that contain both a handle and a key (a handle.backslash.key pair). The key is a very large number (e.g., 128 bits) that uniquely identifies the resource) **Wherein**

- **The processor receives** [column 3, lines 63-65; The OS kernel is inherently operates in the processor] **a request to execute at least one of the plurality of instructions or set of instructions by the code running thereon obtains a second security ID associated with the code,** [column 3, lines 60-62 and column 3, lines 39-41] (As it is disclosed on column 3, lines 60-62, each process requesting the allocation of resource which is defined as concurrently

executing computer programs on column 1, lines 14-15, meets the limitation, a request to execute at least one of the plurality of instructions or set of instructions by the code running thereon obtains resource identifier comprising the handle/key pair that is uniquely identify the resources as explained on column 49-51 **meets the limitation of obtaining a second security ID associated with the process/program/code)**

- **Compares the second security ID with the first security ID, and executes the requested instruction or set of instructions providing that the second security ID matches the first security ID. [Column 4, lines 8-10]** (When a matching key is found, the kernel allows the process to access/executes the requested instruction or set of instructions /resource/program as explained on column 4, lines 8-10)

Therefore the second security ID could be provided to a program which is attempting forgery, however would not be able to access the requested resources since its security ID/indentifier/Key pair would not be the same with the first Security ID which is provided to some other program. In other words application programs as explained on column 23-25 such as word programs and spreadsheet program could have a shared memory but one of the program would be able to access the resource of the other program if and only if it has one and the same key pairs/identifier otherwise it would be denied as explained on column 3, lines 60-column 4, line 11]

Therefore each and every limitations of the independent claims are disclosed by the reference on the record namely **Draves.**

The next argument by the applicant is referring to the dependent claims 4-6, 12-14 and 20-22. Applicant argued that the cited reference fail to provide any suggestion or motivation for modifying the prior art to arrive at Applicant's claimed invention.

Examiner disagrees with this argument, As to the argument made to the motivation, It is not necessary that the reference actually suggest, expressly or in so many words, the changes or improvements that applicant has made. The text for combining references is what the references as a whole would have suggested to one of ordinary skill in the art. See In re Sheckle, 168 USPQ 716 (CCPA 1971) In re McLaghin 170 USPQ 209 (CCPA 1971). In re Young 159 USPQ 725 (CCPA 1968).

Therefore the rejections remains to be valid unless and otherwise the claims are further amended and overcome the rejection without introducing a new matter.

Claim Rejections - 35 USC § 102

3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

4. **Claims 1-3, 7-11,15-19 and 23-24** are rejected under 35 U.S.C. 102(b) as being anticipated by **Richard P. Draves** (hereinafter referred as **Draves**) (U.S. Patent No 5,802,590)

As per claim 9 Draves discloses an apparatus, comprising:

- **A processor for running code thereon**,[column 3, lines 39-42 and column 1, lines 11-22 and figure 2, ref. Num "250"] (As indicated on column 3, lines 39-42, the invention is directed towards a method and system in a kernel of an operating system for providing secure access to computer system

resources. The OS kernel is inherently operates in the processor. And as it is indicated on column 1, lines 39-42, the portion of the operating system that is responsible for the allocation and deallocation of resources is known as the kernel. The kernel interacts with the shell and other programs as well as with the hardware devices on the system, including the processor (also called the central processing unit or CPU), memory and disk drives.)

- **For associating a first security identification (ID) with each of a plurality of instructions or a set of instructions that are to be executed by the processor; [**column 3, lines 43-50 and column 3, lines 60-62**] (**As it is disclosed on column 3, lines 60-62, each process which is defined as concurrently executing computer programs on column 1, lines 14-15, meets the limitation each of a plurality of instructions or a set of instructions are inherently executed by the processor are associated with the resource identifier comprising the handle/key pair that is passed to the process/programs/set of instructions when requesting allocation of resources. Furthermore Draves on column 3, lines 43-50 discloses the following. In a preferred embodiment, the kernel maintains a system-wide resource table that is a hash table and that contains a resource entry corresponding **to each resource allocated by the kernel. The allocated resources are identified by a kernel-generated resource identifier.** The system of the present invention uses resource identifiers that contain both a handle and a key (a handle.backslash.key pair). The key is a very large number (e.g., 128 bits) that uniquely identifies the resource)**.Wherein**

- **The processor receives** [column 3, lines 63-65; The OS kernel is inherently operates in the processor] **a request to execute at least one of the plurality of instructions or set of instructions by the code running thereon obtains a second security ID associated with the code,** [column 3, lines 60-

62 and column 3, lines 39-41] (As it is disclosed on column 3, lines 60-62, each process requesting the allocation of resource which is defined as concurrently executing computer programs on column 1, lines 14-15, meets the limitation, a request to execute at least one of the plurality of instructions or set of instructions by the code running thereon obtains resource identifier comprising the handle/key pair that is uniquely identify the resources as explained on column 49-51 **meets the limitation of obtaining a second security ID associated with the process/program/code)**

- **Compares the second security ID with the first security ID, and executes the requested instruction or set of instructions providing that the second security ID matches the first security ID. [Column 4, lines 8-10]** (When a matching key is found, the kernel allows the process to access/executes the requested instruction or set of instructions /resource/program as explained on column 4, lines 8-10)

5. **As per claim 1,** Claim 1 recites the method version of the independent claim 9 and likewise rejected by the same analogy/ground as that of claim 9.

6. **As per claim 17,** Claim 17 recites the same limitations as that of the independent claim 9 and therefore rejected by the same analogy/ground as that of claim 9.

7. **As per claims 2, 10 and 18, Draves** discloses the method/apparatus/article as applied to claims 1, 9 and 17 above. Furthermore, **Draves** discloses the method/apparatus/article comprising denying the execution of the requested instruction or set of instructions providing that the first and second security IDs mismatch. [Column 4, lines 5-8; figure 8, ref. Num "830"]

8. **As per claims 3, 11 and 19, Draves** discloses the method/apparatus/article as applied to claims 1, 9 and 17 above. Furthermore, **Draves** discloses the method/apparatus/article wherein associating a first security identification (ID) further comprises: storing a first security identification (ID) with each of a plurality of instructions or a set of instructions that are to be executed by a processor.[Column 3, lines 59-62] (The stored resource contains a copy of the key meets the recitation of this claim.)

9. **As per claims 7, 15 and 23, Draves** discloses the method/apparatus/article as applied to claims 1, 9 and 17 above. Furthermore, **Draves** discloses the method/apparatus/article wherein comparing the second security ID with the first security ID further comprises: comparing a portion of the second security ID with a portion of the first security ID.[Column 4, lines 8-10] (A process access for executing the requested instruction or set of instructions or a program or in general accessing the resource is allowed when a match is found by comparing all portions of the first and second identification)

10. **As per claims 8, 16 and 24, Draves** discloses the method/apparatus/article as applied to claims 7, 15 and 23 above. Furthermore, **Draves** discloses the method/apparatus/article wherein executing the requested instruction or set of instructions providing that the second secuity ID matches the first security ID further comprises:

- Executing the requested instruction or set of instructions providing that the portion of the second security ID matches the portion of the first security ID. [Column 4, lines 8-10] (A process access for executing the requested instruction or set of instructions or a program or in general accessing the resource is allowed when a match is found by comparing all portions of the first and second identification)

Claim Rejections - 35 USC § 103

11. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

12. **Claims 4-6, 12-14 and 20-22** are rejected under 35 U.S.C. 103(a) as being unpatentable over **Richard P. Draves** (hereinafter referred as **Draves**) (U.S. Patent No 5,802, 590) in view of **Krueger et al**, (hereinafter referred to as **Krueger**) (U.S. Patent No. 4,962,533)

13. **As per claims 4-6, 12-14 and 20-22, Draves discloses**

- **A processor** [Figure 2, ref. Num "250"] **for running code thereon**,[Column 1, lines 13-14; column 4, lines 16-17] and **for associating a first security identification (ID) with each of a plurality of instructions or a set of instructions that are to be executed by the processor;** [Figure 3, ref. "handle/key"] (As shown on figure 3, for each multiplicity/plurality of processes a handle/key pair is associated.)

Draves does not explicitly discloses

- A first security identification (ID) further comprises:

Classifying at least one instruction or set of instructions from a plurality of instructions that are to be executed by a processor as being security sensitive;

And associating a first security identification (ID) with each of the instructions or set of instructions that are classified as security sensitive.

However, in the same field of endeavor, **Krueger** discloses

Classifying at least one instruction or set of instructions from a plurality of instructions that are to be executed by a processor as being security sensitive and associating a first security identification (ID) with each of the instructions or set of instructions that are classified as security sensitive;[Column 2, lines 43-46; abstract and] (computer system uses security labels for every word in memory and according to the present invention, in a computer system every word in the memory has a corresponding label/security identification. This label indicates the security classification, and compartments if any, of that word of data)

It would have been obvious to one having ordinary skill in the art, at the time the invention was made, to combine the features of having a classification for one instructions/program and associating security identification/label with each instructions or set of instructions as per teachings of **Krueger** in to the method as taught by **Draves**, in order provide a security technique for a computer system in which data retains its classification with a straightforward and reliable mechanism for separating sensitive and non-sensitive data within the system.[see **Krueger** column 2, lines 19-21 and 39-41]

14. The indicated allowability of claims 1-24 is also withdrawn in view of the newly discovered reference(s) to **Kamiya**, Shigeo (hereinafter referred to as **Kamiya**) (U.S. Patent No. 4, 949, 238)

15. **Claims 1-3, 7-11,15-19 and 23-24** are rejected under 35 U.S.C. 102(b) as being anticipated by **Kamiya**, Shigeo (hereinafter referred to as **Kamiya**) (U.S. Patent No. 4, 949, 238)

16. As per claims **Claims 1-3, 7-11,15-19 and 23-24**, **Kamiya** discloses **a method, comprising**

- **Associating a first security identification (ID) with each of a plurality of instructions or a set of instructions that are to be executed by a processor** [Column 2, line 67-column 3, line 10; column 4, lines 49-68; column 5, lines 26-27 and figure 1].(the plurality of branch instructions meets the limitation of plurality of instruction as it is disclosed on column 2, line 67-column 3, line 10. And the true/mask register shown on figure 1, ref. Num "122" meets the limitation of the first security ID.)

- **Requesting to execute at least one of the plurality of instructions of set of instructions by a software code running on the processor;** [Column 5, lines 23-25) (branch instruction executed .....)

- **obtaining a second security ID associated with the software code running on the processor;** (column 3, line 41-42, "the current privilege register")

- **comparing the second security with the first security ID ; [column 3, lines 35-42] and**

- **Executing the requested instruction or set of instructions** [column 2, lines 22-24, "the succeeding microinstruction is normally selected"] **providing that the second security ID matches the first security ID.[column 2, lines 22-24,** ("determined to be true" meets the limitation of the second security ID matches the first security ID)

17. **Claims 4-6, 12-14 and 20-22** are rejected under 35 U.S.C. 103(a) as being unpatentable over **Kamiya**, Shigeo (hereinafter referred to as **Kamiya**) (U.S.

Patent No. 4, 949, 238) in view of **Krueger et al**, (hereinafter referred to as **Krueger**) (U.S. Patent No. 4,962,533)

18. **As per Claims 4-6, 12-14 and 20-22 Kamiya** discloses **a method, comprising**

- **Associating a first security identification (ID) with each of a plurality of instructions or a set of instructions that are to be executed by a processor** [Column 2, line 67-column 3, line 10; column 4, lines 49-68; column 5, lines 26-27 and figure 1].(the plurality of branch instructions meets the limitation of plurality of instruction as it is disclosed on column 2, line 67-column 3, line 10. And the true/mask register shown on figure 1, ref. Num "122" meets the limitation of the first security ID.)

- **Requesting to execute at least one of the plurality of instructions of set of instructions by a software code running on the processor;** [Column 5, lines 23-25) (branch instruction executed .....)

- **obtaining a second security ID associated with the software code running on the processor;** (column 3, line 41-42, "the current privilege register")

- **comparing the second security with the first security ID ; [column 3, lines 35-42] and**

- **Executing the requested instruction or set of instructions** [column 2, lines 22-24, "the succeeding microinstruction is normally selected"] **providing that the second security ID matches the first security ID.[column 2, lines 22-24,** ("determined to be true" meets the limitation of the second security ID matches the first security ID)

Kamiya does not explicitly discloses

- A first security identification (ID) further comprises:

Classifying at least one instruction or set of instructions from a plurality of instructions that are to be executed by a processor as being security sensitive;

And associating a first security identification (ID) with each of the instructions or set of instructions that are classified as security sensitive.

However, in the same field of endeavor, **Krueger** discloses

Classifying at least one instruction or set of instructions from a plurality of instructions that are to be executed by a processor as being security sensitive and associating a first security identification (ID) with each of the instructions or set of instructions that are classified as security sensitive;[Column 2, lines 43-46; abstract and] (computer system uses security labels for every word in memory and according to the present invention, in a computer system every word in the memory has a corresponding label/security identification. This label indicates the security classification, and compartments if any, of that word of data)

It would have been obvious to one having ordinary skill in the art, at the time the invention was made, to combine the features of having a classification for one instructions/program and associating security identification/label with each instructions or set of instructions as per teachings of **Krueger** in to the method as taught by **Kamiya**, in order provide a security technique for a computer system in which data retains its classification with a straightforward and reliable mechanism for separating sensitive and non-sensitive data within the system.[see **Krueger** column 2, lines 19-21 and 39-41]

Conclusion

19. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed

within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Samson B Lemma whose telephone number is 571-272-3806. The examiner can normally be reached on Monday-Friday (8:00 am---4: 30 pm).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, BARRON JR GILBERTO can be reached on 571-272-3799. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

SAMSON LEMMA
S.L.
08/18/2006

GILBERTO BARRON JR.
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100